but to the next sequential one of said blocks thereto;

(3) UPP: rewrite said free-chain pointer in the directory profile to point to said one block and save the old value of said free-chain pointer;

(4) UPF: rewrite the backward pointer in said one block to point to the old leading free-chain block so that said one block now becomes the new leading free-chain block;

(5) UPB: write new entries into the next higher index level to show changes just made to the lower level;

(6) DEL: remove, from said next higher level, the entry which pointed at a block just transferred to said free-chain;

(7) UNB: rewrite the backward pointer in said next sequential block to point to said next previous block;

(8) REM: remove said slot contents from said one block that were shifted into said next previous block by copying in operation SFT.

34. The method as set forth in claim **33** wherein the effects of a generated one of said progress vectors at a level other than the base level are inhibited until the next previous said basic operation UPB and/or DEL is effective in relation to the last occupied one of said slots in said index block at that level.

35. The method as set forth in claim **34** wherein said basic operation SFT is inhibited in relation to any said source block when it is chained to a full said next sequential index block and is, itself, full.

36. The method as set forth in claim **35** wherein said basic operation SFT is further limited to maintain a predetermined number of said slots in said blocks unoccupied by upkeep processing operations.

37. The method as set forth in claim **36** wherein said basic operation SFT is further limited to maintain a predetermined number of said slots in said block at other than said base level, and a greater predetermined number of said slots at said base level, unoccupied by upkeep processing operations.

* * * * *